STIC Scientific and Technical Information Center	Dialog®

Core FT1:

Business & Industry , File 9 (1994 - present)

ABI/INFORM®, File 15 (1971 - present)

Gale Group PROMT®, File 16 (1990 - present)

Gale Group Computer Database , File 275 (full-text 1/1988 - present)

Business Wire, File 610 (Mar 1999 - present)

Business Wire, File 810 (1986 - February 1999)

Core FT2:

Dialog Global Reporter, File 20 (May 1997 - present)

The McGraw-Hill Companies Publications Online, File 624 (1985 - present)

Gale Group New Product Announcements/Plus® (NPA/Plus, File 621 (1985 - present)

Gale Group Newsletter Database , File 636 (1988 - present)

PR Newswire, File 613 (May 1999 - present)

San Jose Mercury News, File 634 (Jun 1985 - present)

PR Newswire, File 813 (May 1987 - May 1999)

Set#	Query
L1	(right ADJ of ADJ way) (road$4 WiTH configur$5) (accident WiTH (description type))
L2	((determin$5 assess$4) WiTH liabilit$4) SAME (accident crash collision)
L3	L1 AND L2
L4	L1 SAME L2
L5	L1 WITH L2

27/9/32 (Item 32 from file: 15)
00611829 92-26932
Expert Systems: The Missing Step

Helland, Joan
Best's Review (Prop/Casualty) v93n1 pp: 34-38, 78
May 1992

CODEN: BRPIDU
ISSN: 0161-7745 Journal Code: BIP
Document Type: Journal article Language: English Length: 4 Pages
Special Feature: Charts
Word Count: 1887

Abstract:

Due to the expertise they can capture and make accessible to many people, expert systems are a valuable decision tool for many businesses. Regardless of how well each step is implemented, building a useful expert system hinges on how well the developer captures an expert's thought process. It is easier to capture the expert's thought process when the developer works with a mental model, which, ideally, is employed after the interview stage but before the development of a design or prototype. The steps for developing an expert system that incorporates mental models include: 1. gathering background information in the expert's field of knowledge, 2. interviewing the expert, 3. developing an initial mental model, 4. developing a design or prototype, and 5. conducting additional interviews, enhancing and refining the mental model, and coding the system. Without a mental model, developers new to an application have difficulty understanding how rules fit together or the consequences of removing or altering a rule.

Text:

Expert systems are a valuable decision tool for many businesses because the expertise they capture can be made accessible to many people. But because these computer systems are a relatively new technology, development methodologies that quickly and accurately capture a person's knowledge are just beginning to appear.

Existing methodologies usually involve gathering background information in the expert's field of knowledge, interviewing the expert, developing a design or prototype, conducting additional interviews and coding the system until sufficient expert knowledge is incorporated into the system. Regardless of how well each step is implemented, building a useful expert system hinges on how well the developer captures an expert s thought process. Unraveling an expert's thoughts becomes increasingly difficult as the area of expertise becomes more complex. This is especially true in the insurance industry when one considers the complex judgments made by professionals in areas like claims and underwriting.
Capturing the complexity of an expert's thought process is easier when the developer works with a mental model. Mental models, which are used in cognitive psychology to describe the organization of mental activities, ideally are employed after the interview stage but before the development of the design or prototype. Having a mental model at this point guides the total system development and helps ensure that the final system delivers the anticipated expertise.

However, even with an explicit step for a mental model, the challenge for the systems developer remains the same: figuring out the thought process, or mental model, that an expert uses to make a decision. This is difficult because experts are often "data driven" rather than procedural. For example, the insurance claims expert may begin working on a bodily-injury claim by searching the claim file for information related to **liability** and injuries. In the first file, the expert focuses on the **accident description** to help **determine** **liability** and then studies medical reports to understand the

injuries. After identifying a concern in how the **accident** caused the injuries, he or she returns to the **accident description** for clarification. In the next file, the problems and concerns identified are different, and therefore, the order in which the file is reviewed is different.

RECURRING THEMES

The claims expert's approach is called data-driven because the data found in the initial review of the file "drives" the expert to look to a particular place for additional data, and what is found there "drives" the expert to the next place in the file. Since the data is different from file to file, the order in which the data is reviewed also is different. This type of thought process makes sorting out the organization and structure of the mental model difficult.

Even though an expert may be data-driven, it is possible to develop a mental model because the expert focuses on recurring ideas or themes in every file. In the claims situation, the major components making up the liability and injury assessments are reviewed at some point in every file. In another example, financial analysis, the components making up working capital, net worth and profitability are discussed by the expert in every file, although in varying order from file to file. These recurring ideas or themes make up the organization and structure of the mental model.

MODELS OF CHESS MASTERS

Research with chess masters demonstrates how mental models work. In an exercise described by A.D. de Groot in Thought and Choice in Chess, psychologists asked chess masters and amateurs to study a chess board for five seconds, with the pieces positioned as if in a game. They were then asked to re-create the board from memory. The chess masters correctly placed about 20 pieces, while the amateurs could remember only seven pieces.

To understand what gave the chess masters the advantage, the psychologists rearranged the pieces on the board so that they were no longer in the middle of a game. Under these conditions, the chess masters and amateurs both remembered only about seven pieces. Rearranging the pieces disrupted the chess masters ability to organize them into meaningful units, or abstractions. The chess masters' abstractions used to organize the pieces represent the chess master's mental model.

Subsequent research showed more about how chess pieces make up an abstraction. In a study described by W.G. Chase and H.A. Simon in their article, "Perception in Chess," which appeared in Cognitive Psychology in 1973, psychologists removed the time constraint and allowed the participants to look back at the original pieces. Both groups looked at the original board and placed two or three pieces and then looked back again and placed another two or three pieces. However, the chess masters spent significantly less time looking back. This indicates that experts could group the two or three pieces into a meaningful abstraction faster than could the amateurs.

Using the chess research as a basis for understanding mental models, consider the thought process used in underwriting contract surety bonds. The underwriting file has a huge amount of information, including financial statements, work-in-progress details, prior-bid results, information on previous work experience of the contractor and a myriad of other details. The data in the file are like the pieces on the chess board. The expert underwriter reviews the file by combining certain data elements into the meaningful units, or abstractions, which are expressed during the interviews with the experts as recurring ideas and themes. Those

abstractions are combined into higherlevel abstractions making up the mental model.

Figure 1 shows the lower-level abstractions constructed by the expert in the financial analysis aspect of underwriting surety bonds. (Figure 1 omitted) The data for operating income for three years are combined into the "operating income abstraction. The underwriter reviews the amount and trend in the figures, as well as other information related to the contractor to make the judgment that the operating income for this particular contractor is poor. A similar abstraction is constructed around net income before taxes. These two abstractions and others are used to construct a higher-level abstraction of profitability.

Figure 2 shows more of the same mental model. (Figure 2 omitted) It has the high-level abstraction for profitability and two other high-level abstractions of working capital and net worth. These two additional abstractions have their own complex lower-level abstractions, which, for simplicity, are not shown. The purpose of Figure 2 is to see how the high-level abstractions are combined into the financial-strength abstraction. Financial-strength and other high-level abstractions, such as technical strength, are combined to form the final underwriting decision.

This example shows how an expert underwriter reviews the data in a file, as a chess master reviews the pieces on the board, and then organizes and structures the data into abstractions. The mental model describes the transformation of data into lower-level abstractions and then shows how lower-level abstractions combine to form higher level abstractions. Just as chess masters transform chess pieces into meaningful groups, skilled underwriters transform insurance data into expert abstractions that are used to make expert decisions.

Mental models often are confused with decision trees because their diagrams appear to be similar. Figure 3 shows the financial aspects of the surety mental model redrawn as a decision tree. (Figure 3 omitted) The tree imposes a procedure to the decision since the profitability is analyzed first, and then net worth is analyzed in the second series of branches, then working capital, and so on. Ultimately, after making the analysis in the specified order, the final decision is derived.

Notice that the mental model in Figure 2 does not impose an order on the abstractions used to derive the underwriting decision. This reflects the nonprocedural, data-driven nature of underwriting decisions. Because insurance experts tend to be data driven imposing the rigid order of a decision tree on their thought process can be disconcerting to the expert and misleading to the developer.

The steps for developing an expert system that incorporates mental models include: gathering background information in the expert's field of knowledge; interviewing the expert; developing an initial mental model; developing a design or prototype; and conducting additional interviews, enhancing and refining the mental model, and coding the system. The initial mental model explicitly captures the expert's thought process before any code is written. As suggested, this is never simple but is easier if developers expect experts to be data-driven, expect to hear recurring ideas and themes that relate to the abstractions in the thought process and then use the abstractions to organize and structure what they have heard into a mental model.

Deriving the initial model assists with design decisions. For example, consider the decision on what to include in an expert system. Since no system does everything, developers and experts decide what to include. In the surety example, Figure 2 shows the financial-strength abstraction and

technical-strength abstraction coming together to form the final underwriting judgment.
Technical strength refers to the contractors' experience in the type of construction work under way and their ability to perform work on schedule and within budget. Based on the mental model, the financial strength and technical strength are two relatively separate abstractions, and the system does not necessarily have to provide assistance for both areas. In fact, the users could decide the system will initially support only the financial aspects and not the technical strength. Later enhancements to the system could address technical strength.

The rationale for the scoping decision might be that if financial-strength analysis is more error-prone, underwriters need more assistance with it than with technical strength. Alternatively, the abstractions making up expertise for financial strength are more standardized, so this area is easier for the experts to verbalize than is technical strength. Both reasons might provide rationale for scoping the initial phases of the system design to include financial strength and exclude technical strength.

The initial model also affects the additional interviews and the coding. The mental model serves to focus the interviews. With a mental model to guide the process, the developer can more easily identify whether vaguely expressed ideas or themes are restatements of earlier abstractions or new abstractions. In addition, when an expert uses a heuristic, or rule of thumb, the developer more easily recognizes where it applies and why it is significant.
In the context of a mental model, the expert's rules of thumb can be organized by the abstractions. There are rules for transforming data into low-level abstractions, making judgements on abstractions such as "poor" operating income in the surety example and transforming lower-level abstractions into higher-level abstractions. For instance, a rule might be that a file with strong financials (working capital and net worth) and poor profitability has acceptable financial strength. There is a different rule for the reverse, a file with poor financials and strong profitability has unacceptable financial strength.

When the system is in use, the mental model will help with maintenance. Without a mental model, developers new to an application have little idea of the "big picture." They have difficulty understanding how the rules fit together or what the consequences are of removing or altering a rule. With a mental model to study as part of the system design, developers can proceed directly to rules that affect the abstractions in need of revision. Maintenance is never easy, but mental models provide guidance on how and where to initiate changes.

Joan Helland, Ph.D., is director of application services for CIGNA Companies, Philadelphia.

THIS IS THE FULL-TEXT.

Geographic Names: US

Descriptors: Insurance industry; Expert systems; Systems development; Models; Recommendations; Insurance claims
Classification Codes: 9190 (CN=United States); 8220 (CN=Property & casualty insurance); 5240 (CN=Software & systems)